AUTOSUBMIT: A VERSATILE TOOL TO MANAGE WEATHER AND CLIMATE EXPERIMENTS IN DIVERSE SUPERCOMPUTING ENVIRONMENTS

D. Manubens[1], J. Vegas [1] , F. Doblas-Reyes [1,2,3]

[1] Institut Catalá de Ciències del Clima (IC3) [2] Barcelona Supercomputing Center (BSC) [3] Institució Catalana de Recerca i Estudis Avanzats(ICREA)

1. What is Autosubmit ?

Autosubmit is a python-based tool for submitting, managing and monitoring weather and climate simulations. It was developed at IC3 to fill an uncovered need at the moment: allow scientists to run experiments easily on different supercomputers and to reduce the time they spend monitoring model runs.

Autosubmit is currently used at IC3 to run EC-Earth and NEMO models and at Barcelona Supercomputing Centre (BSC) to run NMMB air quality model.

Autosubmit has been used to manage models running at supercomputers in IC3, BSC, ECMWF, EPCC, PDC and OLCF.

Autosubmit 3.0 beta version is now available via PyPi package under the terms of GNU General Public License. Slight changes in the characteristics described in this poster could be found in the final stable release expected in May 2015.

2. Why is Autosubmit needed ?

Autosubmit is the only existing tool that satisfies the following requirements from the weather and climate community:

- **Automatisation**: Job submission to machines and dependencies between jobs are managed by Autosubmit. No user intervention is needed.
- **Data provenance**: Assigns unique identifiers for each experiment and stores information about model version, experiment configuration and computing facilities used in the whole process.
- **Failure tolerance**: Automatic retrials and ability to rerun chunks in case of corrupted or missing data.
- **Resource management:** Autosubmit manages supercomputer particularities, allowing users to run their experiments in the available machine without having to adapt the code. Autosubmit also allows to submit tasks from the same experiment to different platforms.

3. How does Autosubmit work ?

Experiment creation

To create a new experiment, run the command "autosubmit expid –h *HPCname* –d *Description*". *HPCname* is the name of the main HPC platform for the experiment: it will be the default platform for the tasks. *Description* is a brief experiment description.

This command assigns a unique four character identifier (xxxx) to the experiment and creates a new folder in experiments repository with structure shown in Figure 1.

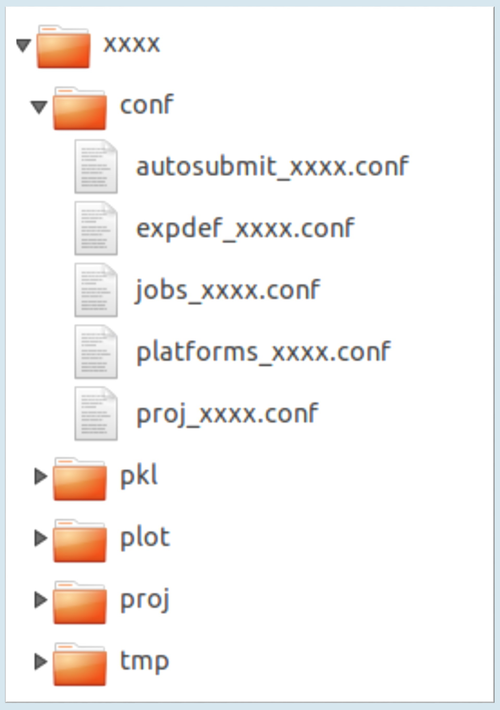

Figure 1: Example of an experiment directory tree.

Experiment configuration

To configure the experiment, edit expdef_xxxx.conf, jobs_xxxx.conf and platforms_xxxx.conf in the conf folder of the experiment.

After that, the user is expected to run "autosubmit create xxxx". This command creates the experiment project in the proj folder. The experiment project contains the scripts specified in jobs_xxxx.conf and a copy of model source code and data specified in expdef_xxxx.conf.

Start dates, members and chunks (number and length).

Experiment project source: origin (version control system or path) and project configuration file path.

expdef_xxxx.conf

Workflow to be run: scripts to execute, dependencies between tasks, task requirements (processors, wallclock time...) and platform to use.

jobs_xxxx.conf

HPC, fat-nodes and supporting computers configuration.

Usually provided by technicians, users will only have to change login and accounting options for HPCs.

platforms_xxxx.conf

Project dependant experiment variables that Autosubmit will substitute in the scripts to be run.

proj_xxxx.conf

Experiment run

To run the experiment, just execute the command "autosubmit run xxxx". Autosubmit will start submitting jobs to the relevant platforms (both HPC and supporting computers) by using the scripts specified in jobs_xxxx.conf. Autosubmit will substitute variables present on scripts where handlers appear in %*variable_name*% format. Autosubmit provides variables for current chunk, start date, member, computer configuration and more., and also will replace variables form proj_xxxx.conf.

To monitor the status of the experiment, the command "autosubmit monitor xxxx" is available. This will plot the workflow of the experiment and the current status.

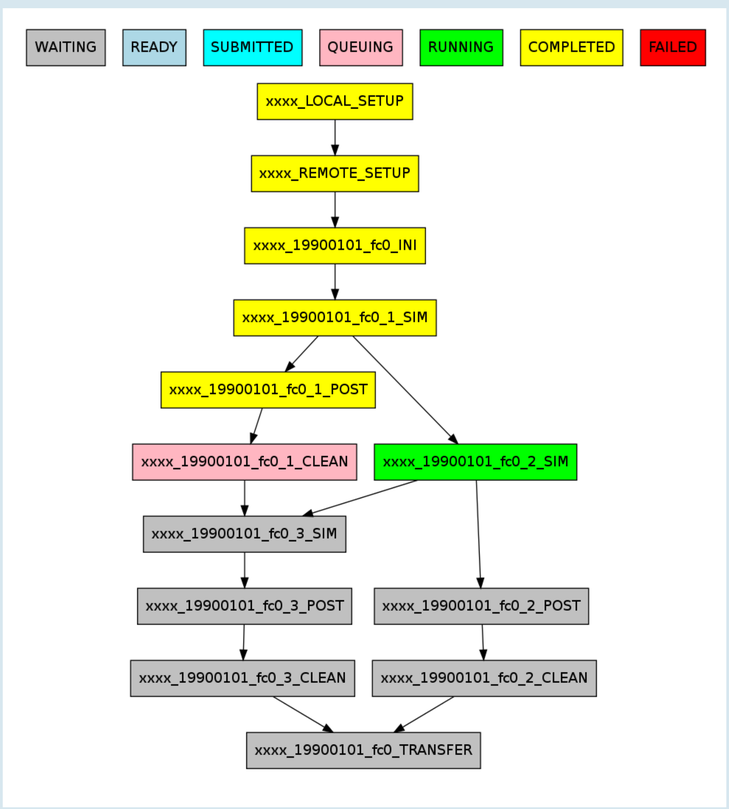

Figure 2: Example of monitoring plot for EC-Earth run with Autosubmit for 1 start date, 1 member and 3 chunks.

4. Autosubmit and EC-Earth

Autosubmit is used at IC3 to run EC-Earth. To do that, a git repository has been created that contains the model source code and the scripts used to run the tasks. The workflow is defined using seven job types, as shown in figure 2. These job types are:

- **Local_setup**: prepares a patch for model changes and copies it to HPC
- **Remote_setup**: creates a model copy and applies the patch to it
- **Ini**: prepares model to start the simulation of one member
- **Sim**: runs a simulation chunk (usually 1 to 3 months)
- **Post**: post-process outputs for one simulation chunk
- **Clean**: removes unnecessary outputs from the simulated chunk
- **Transfer**: transfers post-processed outputs to definitive storage

5. Future work

Plans for further development of Autosubmit include:

- Expand data stored in database to store all data needed for CMIP6.
- Create a wrapper for jobs that allows to send various independent tasks as one big job to reduce queueing time and to allow participation in tier-0 projects.
- Evaluate the suitability of using Cylc (http://cylc.github.io/cylc) for workflow configuration to converge with community standards.

Acknowledgements

The work leading to these results has received funding from the EU Seventh Framework Programme FP7(2007-2013) under grant agreements n° 308378 (SPECS) and n° 312979 (IS-ENES2), and the Catalan Government.